

AMENDMENTS TO THE CLAIMS:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A data store query system comprising:

a data store that includes a collection of records;

a constant-sized sorted result buffer; and

a query interface operable to operate to:

receive a limit and order query that includes both of an order criteria and a limit criteria, the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query,

~~fill to fill~~ the sorted result buffer with a first N number of records from the data store, to iteratively order the sorted result buffer based upon the order criteria,

iteratively identify the last record of the sorted result buffer, iteratively modify the query to include a where criteria that compares remaining records in the data store to the identified last record based on a sort order determined by the order criteria,

iteratively compare the remaining records in the data store against [[a N th]] the identified last record in the sorted result buffer based upon the order criteria and the where criteria,

iteratively to iteratively replace the [[N th]] identified last record in the sorted result buffer with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the [[N th]] identified last record in the sorted result buffer, and output to output the sorted result buffer as the result set of records.

2. (Original) The data store query system of claim 1 wherein the data store is a database or a fast cache.

3. (Previously Presented) The data store query system of claim 1 wherein the collection of records further comprises a table having an attribute, and wherein the query interface is operable to receive the limit and order query placing order constraints on the attribute.
4. (Original) The data store query system of claim 1 wherein the query interface creates a revised sorted result buffer in response to a modification of the limit and order query, the modification being made during a pause in execution of the limit and order query.
5. (Original) The data store query system of claim 1 wherein the sorted result buffer is stored in random access memory.
6. (Original) The data store query system of claim 1 wherein the query interface is operable to receive the limit and order query formulated using standard query language (SQL).
7. (Original) The data store query system of claim 1 wherein the query interface is operable to receive the limit and order query that requests the first or last N records satisfying the query.
8. (Original) The data store query system of claim 1 wherein the query interface is operable to identify data in the data store that satisfies the limit and order query using the sorted result buffer by iteratively reformulating the limit and order query until the sorted result buffer contains data satisfying the limit and order query.
9. (Currently Amended) A method for satisfying limit and order queries including:
 - receiving a limit and order query that includes both of an order criteria and a limit criteria, the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query;
 - filling a constant-sized sorted result buffer with a first N number of records from a data store;
 - iteratively ordering the sorted result buffer based upon the order criteria;

iteratively identifying the last record of the sorted result buffer,
iteratively modifying the query to include a where criteria that compares remaining
records in the data store to the identified last record based on a sort order determined by the order
criteria,

iteratively comparing the remaining records in the data store against [[a Nth]] the
identified last record in the sorted result buffer based upon the order criteria and the where
criteria;

iteratively replacing the [[Nth]] identified last record in the sorted result buffer with a
remaining record in the data store based upon iteratively comparing remaining records in the data
store against the [[Nth]] identified last record in the sorted result buffer; and

outputting the sorted result buffer as the result set of records.

10. (Previously Presented) The method of claim 9 wherein the limit and order query is formulated using standard query language (SQL).

11. (Previously Presented) The method of claim 9 wherein filling the sorted result buffer with the first N number of records from the data store includes:

scanning the data store without consideration of the order criteria to identify records otherwise satisfying the limit and order query; and

placing identified records into the sorted result buffer until the sorted result buffer includes the maximum number of records specified by the limit criteria.

12. (Previously Presented) The method of claim 9 wherein the limit and order query requests the first N records satisfying the order criteria.

13. (Previously Presented) The method of claim 9 wherein the limit and order query requests the last N records satisfying the order criteria.

14. (Currently Amended) An apparatus comprising a storage medium having instructions stored thereon, the instructions including:

a first code segment for receiving a limit and order query that includes both of an order criteria and a limit criteria, the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query;

a second code segment for filling a constant-sized sorted result buffer with a first N number of the set of data records from a data store;

a third code segment for iteratively ordering the sorted result buffer based upon the order criteria;

a fourth code segment for iteratively identifying the last record of the sorted result buffer;

a fifth code segment for iteratively modifying the query to include a where criteria that compares remaining records in the data store to the identified last record based on a sort order determined by the order criteria;

a sixth code segment for iteratively comparing the remaining records in the data store against a N th record in the sorted result buffer based upon the order criteria and the where criteria;

a fifth seventh code segment for iteratively replacing the [[N th]]identified last record in the sorted result buffer with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the [[N th]]identified last record in the sorted result buffer; and

an eighth a-sixth code segment for outputting the sorted result buffer as the result set of records.

15. to 20. (Cancelled)

21. (Previously Presented) The data store query system of claim 1, wherein the size of the constant-sized sorted result buffer is based on the limit criteria.

22. (Previously Presented) The method of claim 9, wherein iteratively replacing the N th record in the sorted result buffer the N th record further comprises iteratively replacing the last record in the sorted memory buffer.

23. (Previously Presented) The method of claim 9, wherein iteratively ordering the sorted result buffer further comprises ordering the sorted result buffer for each iterative replacement of the N th record.

24. (Currently Amended) A method for satisfying limit and order queries including:
receiving a standard query language (SQL) formatted limit and order query that includes both of an order criteria and a limit criteria, the limit and order criteria requesting the first or last N records satisfying the order criteria, and the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query;

filling a constant-sized sorted result buffer with a first N number of records from a data store of a customer relationship management system, an enterprise resource planning system or a supply chain management system, the data store comprising a database or a fast cache, wherein filling the sorted result buffer with the first N number of records from the data store further comprises:

scanning the data store without consideration of the order criteria to identify records otherwise satisfying the limit and order query, and

placing identified records into the sorted result buffer until the sorted result buffer includes the maximum number of records specified by the limit criteria;

iteratively ordering the sorted result buffer based upon the order criteria;

iteratively identifying the last record of the sorted result buffer,

iteratively modifying the query to include a where criteria that compares remaining records in the data store to the identified last record based on a sort order determined by the order criteria,

iteratively comparing the remaining records in the data store against a N th record in the sorted result buffer based upon the order criteria and the where criteria;

iteratively replacing the [[N th]] identified last record in the sorted result buffer with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the [[N th]]identified last record in the sorted result buffer; and

outputting the sorted result buffer as the result set of records.

25. (New) The method of claim 9, wherein execution of the query is paused when the query is modified.
26. (New) The method of claim 9, wherein the execution of the query is resumed at the current table location after the query is modified.
27. (New) The method of claim 9, wherein the sort order is a "less than" sort order.